REMARKS

Claims 1-2, 5-20, 30-32, and 34-37 are pending. Claims 3-4, 21-29, and 33 have been canceled. Claims 1, 14, 30, and 31 have been amended. New claims 36-37 have been added.

Summary of Examiner Interview

On May 11, 2006, the MacDonald and Gibson references were discussed in relation to the pending claims with the Examiner. In particular, MacDonald's disclosure or lack thereof of the storage of user files within NVSD 7, and the disclosure within MacDonald of usage of *another* device for such file storage after the CPU was started up were discussed. As the Examiner stated in the Interview Summary dated May 15, 2006, agreement was not reached. However, it was agreed that Gibson does not teach all the elements of any of the pending claims.

Claim Rejections Under 35 U.S.C. §102

Claims 1-3, 6-9, 14, 16-18, 20, 22-24, 26-28 and 31-32 are rejected under 35 U.S.C. §102(a) as being anticipated by WO 01/52062 to Gibson et al. ("Gibson").

As discussed with the Examiner during an interview of May 11, 2006, Gibson teaches storing boot code on a type of code storage device at page 3, lines 30-37. As discussed previously, a code storage device is distinguished from an application and file storage device in the application text and figures. This has been discussed at length in the course of the current prosecution and as the Examiner has acknowledged this distinction in the interview, as seen in the Interview Summary dated May 15, 2006, no further discussion of Gibson is warranted at this point.

Gibson was also used as the basis of rejection under § 103 together with other references, but as Gibson does not teach the claimed application and file storage device, and the claimed usage thereof, Gibson, alone or in combination with the other references, does render any of the pending claims obvious under § 103.

Claim Rejections Under 35 U.S.C. §103

The Examiner found the previous Response unpersuasive and reiterated his rejection dated October 31, 2005 in the Office Action dated February 22, 2006, to which this is in response.

While many of the claims have currently been amended in order to reach agreement and facilitate allowance of the application, it is respectfully reiterated that the Examiner's interpretation of the teachings of MacDonald is inaccurate and overbroad. The specification of the current application describes and sheds light on the intended meaning of an application and file storage device (AFSD). MacDonald does not teach the storage of user files on NVSD 7, which the Examiner equates with the claimed AFSD. To the contrary, MacDonald teaches that another (un-described) non volatile storage device is used to store user files. *See* MacDonald at page 3 lines 1-2. One of skill in the art recognizes that storage of and access to user files takes place *after* the CPU is started up, and MacDonald expressly teaches that "other non volatile storage devices" are used for this purpose *after* the CPU is started up. *See* Id. As MacDonald is directed strictly to startup, it is logical that the overall architecture of the larger system, including devices utilized after startup, would not be disclosed or taught by MacDonald. The Examiner's assertion that because the type of drive that can function as the claimed AFSD is disclosed by MacDonald, that this claim element and the associated limitations are taught by MacDonald is not well taken.

In light of the current amendments, this point should not be an issue. However, should an appeal be merited this point may be elaborated further.

Moving on to the currently pending claims, claim 1, as amended, is reproduced below.

1. (Currently Amended) A system for starting and controlling operation of an intelligent device comprising:

a NAND flash memory application and file storage device configured to read and write data files, one or more of the data files including the basic input/output system (BIOS) interface;

a random access memory (RAM);

a loading logic circuit that configures the application and file storage device to operate and,

places the chip enable signal of the application and file storage device in an active state;

sends a command to the application and file storage device over a data bus of the device;

awaits the application and file storage device to change state from a busy state, while processing the command, to a ready state;

sets the command latch enable and read address latch enable signal to an inactive state;

sends a read enable signal to the application and file storage device; and

sends a write signal to the RAM,

and thereby copies a portion of the BIOS from the storage device into the RAM.

Macdonald does not teach the limitation of "a NAND flash memory application and file storage device configured to read and write data files, one or more of the data files including the basic input/output system (BIOS) interface." Nor does MacDonald teach "a loading logic circuit that configures the application and file storage device to operate and, places the chip enable signal of the application and file storage device in an active state; sends a command to the application and file storage device over a data bus of the device; awaits the application and file storage device to change state from a busy state, while processing the command, to a ready state; sets the command latch enable and read address latch enable signal to an inactive state; sends a read enable signal to the application and file storage device; and sends a write signal to the RAM, and thereby copies a portion of the BIOS from the storage device into the RAM."

These numerous specific limitations are directed towards enabling and directly controlling the intricacies of the storage device with the loading logic circuitry. Reading of files is generally processor controlled. In all complete hard disk drives or other storage drives such as a CD or tape drive, there is a processor integrated into the device that pre-selects where data is coming from. With NAND flash memory, there is no built in processing power. Therefore, it would not be necessary to have such a loading logic circuit in a system incorporating such a complete drive. Furthermore, a complete or standard drive with a built in processor would not function properly with the claimed loading logic circuitry.

One analogy that may be helpful is that of a hard disk drive. A hard disk drive incorporates hardware (including a platter, and an arm with a head) and a controller or processor. The claimed characteristics of the loading logic circuit are meant to operate directly with the

NAND flash memory. In the analogy of the hard disk drive, this would equate to directly controlling and operating the hardware (platter, and arm/head).

MacDonald, alone or in combination with the admitted prior art, or any of the cited references does not teach controlling file storage operations of NAND flash memory with the claimed loading logic circuit. The DMAC of MacDonald does not have the claimed characteristics of the loading logic circuit. It does not configure or control the intricacies of storing and retrieving data to/from the NVSD of MacDonald. Such aspects of the present invention are simply not taught by and outside the scope of MacDonald. Furthermore, one of skill in the art would understand that the DMAC disclosed by MacDonald only "knows" what port it can read data from, not how to configure and operate the NVSD at the claimed level. This is clear from the limited disclosure in MacDonald relating to the Power on Values, POV1, POV2, and POV3 that specify only the destination address, description of the NVSD, and transfer quantity respectively. *See* page 5 line 29, to page 6 line 10 of MacDonald.

Therefore, it is submitted that independent claim 1, and all the claims that depend therefrom is in compliance with §102 and §103 and is in condition for allowance.

Independent claim 14 is also not anticipated or rendered obvious by the cited prior art and is reproduced below.

14. (Currently Amended) A method of starting a smart device comprising:
resetting operation of a microprocessor; and thereafter
suspending operation of the microprocessor; and thereafter
enabling operation of an application and file storage device with system level circuitry not provided as part of the application and file storage device; and thereafter
copying a portion of a BIOS from the application and file storage device into RAM with said system level circuitry; and thereafter
starting operation of the microprocessor.

Independent claim 30 is also not anticipated or rendered obvious by the cited prior art and is reproduced below.

30. (Currently Amended) A method of providing an interface between an operating system and hardware devices comprising:
storing the interface in an application and file storage device; and thereafter
copying the interface from the application and application and file storage device into RAM without using a microprocessor by:

enabling the application and file storage device and the RAM; and thereafter

enabling an address counter to output a value; and thereafter

correlating the value with a RAM address; and thereafter

sending data from the application and file storage device over a data bus to the RAM address; and thereafter

verifying the sent data with error correction code, and

incrementing the address counter.

Independent claim 31 is also not anticipated or rendered obvious by the cited prior art and is reproduced below.

31. (Currently Amended) A system for booting a microprocessor controlled device comprising: an application and file storage device having a plurality of files, said application and file storage device not incorporating a controller for operating the device;

a random access memory;

a microprocessor;

human interface devices; and

an interface for communicating between the microprocessor, the application and file storage device and the human interface devices, the interface residing in a file of the file storage device; and means for copying a portion of the interface into the random access memory, said means functioning to directly control the read and write operations of the application and file storage device.

New claims 36 and 37 are likewise not anticipated or rendered obvious by the cited prior art and are reproduced below. New claims 36 and 37 are fully supported by the specification and do not introduce new matter.

36. (New) A method of operating a smart device comprising:

providing a general purpose application and file storage device for the storage of user files;

allocating a small portion of the general purpose application and file storage device for a BIOS of the device;

resetting operation of a microprocessor; and thereafter

suspending operation of the microprocessor; and thereafter

copying a portion of a BIOS from the application and file storage device into RAM by

enabling the general purpose application and file storage device and the RAM; and thereafter

enabling an address counter to output a value; and thereafter

correlating the value with a RAM address; and thereafter

sending data from the application and file storage device over a data bus to the RAM address; and thereafter

incrementing the address counter; and thereafter

starting operation of the microprocessor; and

using the general purpose application and file storage device to read and write user files, and during such usage, preventing the BIOS from being overwritten.

37. (New) A method of starting and operating a processor controlled system comprising an operating system, hardware devices, and an interface between the operating system and the hardware devices, the method comprising:

storing the interface in a directly accessed NAND flash application and file storage memory; and thereafter

copying the interface from the application and file storage memory into RAM by:

enabling the application and file storage memory and the RAM; and thereafter

enabling an address counter to output a value; and thereafter

correlating the value with a RAM address; and thereafter

sending data from the application and file storage memory over a data bus to the RAM address; and thereafter

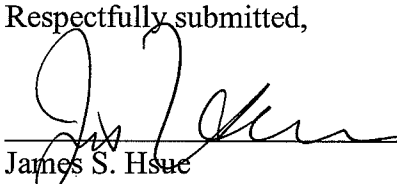
incrementing the address counter.

Therefore, it is submitted that all the pending claims are not anticipated or rendered obvious and are in condition for allowance

Conclusion

Accordingly, it is believed that this application is now in condition for allowance and an early indication of its allowance is solicited. However, if the Examiner has any further matters that need to be resolved, a telephone call to the undersigned attorney at 415-318-1162 would be appreciated.

Respectfully submitted,



May 24, 2006

James S. Hsue
Reg. No. 29,545

Date

PARSONS HSUE & DE RUNTZ LLP
595 Market Street, Suite 1900
San Francisco, CA 94105
(415) 318-1160 (main)
(415) 318-1162 (direct)
(415) 693-0194 (fax)